IN THE CLAIMS:

1-19. canceled

20. (currently amended) In a switch system including a first plurality of crossbars with a plurality of parallel routed switch inputs and a plurality of parallel routed switch outputs, a hierarchical arbitration method comprising:

establishing a crossbar counter list;

accepting variably sized information packets including a plurality of cells, at a plurality of switch inputs, the plurality of information packets addressing a plurality of switch outputs;

simultaneously arbitrating for a plurality of links between switch inputs and switch outputs in a plurality of arbitration cycles for each crossbar, where each switch output in a crossbar simultaneously nominates an available switch input;

locking the links; [[and,]]

transferring information packets across the links; and,

wherein simultaneously arbitrating for a plurality of links between switch inputs and switch outputs, for each of the first plurality of crossbars, includes arbitrating between a plurality of available switch inputs, in response to the crossbar selected from the crossbar counter list.

21. canceled

22. (currently amended) The method of claim [[21]] 20 wherein simultaneously arbitrating for a plurality of links, for each of

the first plurality of crossbars, includes arbitrating in a plurality of arbitration cycles, for each selected crossbar.

23. (original) The method of claim 22 wherein simultaneously arbitrating for links to each switch output, for each of the first plurality of crossbars includes:

nominating first available switch inputs for each switch output for a first crossbar, selected in response to the crossbar counter;

following the acceptance of the first available switch inputs for each switch output, setting each available switch input priority list pointer to a second switch input, next in sequence to the first switch input;

setting the crossbar counter to a second crossbar, next in succession to the first crossbar; and,

nominating switch inputs closest in succession to the second switch input for each switch output for the second crossbar.

24. (original) The method of claim 23 wherein simultaneously arbitrating for a plurality of links, for each of the first plurality of crossbars, includes:

accepting first nominating switch outputs for a first crossbar, for each switch input;

following the acceptance of the first nominating switch outputs for each switch input, setting each nominating switch output priority list pointer to a second switch output, next in sequence to the first switch output;

setting the crossbar counter to a second crossbar, next in succession to the first crossbar; and,

accepting nominating switch outputs in the second crossbar closest in succession to the second switch output for each switch input.

25. (currently amended) In a switch system including a first plurality of crossbars with a plurality of parallel routed switch inputs and a plurality of parallel routed switch outputs, a hierarchical arbitration method comprising:

accepting variably sized information packets including a plurality of cells at a plurality of switch inputs, the plurality of information packets addressing a plurality of switch outputs;

at each switch input, queuing the information packets into a plurality of queues;

simultaneously arbitrating for a plurality links between switch inputs and switch outputs;

locking the links;

selecting a queue for each locked link, for each crossbar, by selecting the least recently available queue as follows:

for each switch input, establishing a queue list with a queue pointer;

at a first crossbar, selecting a first queue for each switch input accepting a nominating switch output, in response to the queue pointer;

following the selection of the first queue for each switch input accepting a nominating switch output in the first

crossbar, setting each queue list pointer to a second queue, next in sequence to the first queue; and,

selecting a queue closest in succession to the second queue for each switch input accepting a nominating switch output in a second crossbar; and,

transferring information packets across the links.

26-27. canceled

28. (original) The method of claim 25 wherein accepting information packets, at a plurality of switch inputs, includes accepting information packets having a ranked class of service (COS);

wherein queuing information packets into a plurality of queues includes queuing the information packets by COS; and,

wherein selecting a queue includes each nominated switch input selecting a queue in response to COS of the information packets available for each crossbar.

29. (original) The method of claim 28 further comprising:

establishing a plurality of selection cycles; and,

simultaneously analyzing information packets at the head of each queue in each selection cycle.

30. (original) The method of claim 29 wherein simultaneously analyzing the information packets at the head of each

queue includes analyzing information packets in response to the number of cells in each information packet.

31. (original) The method of claim 30 further comprising:

selecting an accumulation increment for each of the plurality of COS queues, where each accumulation increment corresponds to a selected number of cells; and,

wherein simultaneously analyzing information packets includes comparing the number of cells in the information packet at the head of the queue to its corresponding accumulation increment.

32. (original) The method of claim 31 wherein selecting an accumulation increment for each of the plurality of COS queues includes selecting accumulation increments with larger numbers of cells for higher ranking COS queues.

33. (original) The method of claim 32 wherein simultaneously analyzing information packets includes comparing the number of cells in the information packet at the head of the queue to a corresponding total accumulation in a plurality of selection cycles.

34. (original) The method of claim 33 wherein selecting a queue for a locked link includes:

for each COS queue, establishing a bank for banking accumulation increments;

in each selection cycle, comparing the number of cells in the information packets at the head of each COS queue to a total accumulation that includes the accumulation increment, plus the banked accumulation;

if an information packet has a number of cells less than, or equal to, the total accumulation, making the information packet eligible for selection;

if information packets are eligible from a plurality of queues, picking the information packet in the queue in response to a priority system;

if an information packet is picked, banking the total accumulation, minus the number of cells in the selected packet; and,

if no information packets are picked, banking the total accumulation in each COS queue.

35. (original) The method of claim 34 wherein information packets in the queue are picked in response to a priority system selected from the group including highest COS and next from an ordered service list.

36-42. canceled

43. (currently amended) An hierarchical arbitration system for transferring information across a switch, the system comprising:

a switch including a first plurality of crossbars having a plurality of parallel routed inputs and a plurality of parallel routed

outputs connected to the switch inputs in response to the arbitration commands accepted on a control input;

an arbiter having an output connected to the switch control input to supply simultaneously arbitrated link commands for a plurality of links between the switch inputs and the switch outputs, for each crossbar;

a queue assembler having a plurality of inputs to accept variably sized information packets having a plurality of cells and addressing a plurality of outputs, and a control input to accept queue selection commands, the queue assembler grouping the information packets by switch output address, queuing the information packets into a plurality of queues for each address grouping, and supplying queues, selected in response to queue selection commands, at a plurality of outputs connected to corresponding switch inputs; and,

wherein the switch locks the links between switch inputs and switch outputs, in response to commands from the arbiter, to transfer information packets across the links.

wherein the arbiter includes a crossbar counter to select a crossbar, and in response to a selected crossbar, simultaneously arbitrates between each arbitrating switch output and nominated switch inputs from an available switch input priority list, by selecting the least recently used available switch input in a plurality of arbitration cycles for each crossbar, as follows:

the arbiter nominates the highest priority available switch input, for each arbitrating switch output, in a first arbitration cycle, and if the nominating switch output is not

accepted, the arbiter nominates successively lower priority available switch inputs in subsequent arbitration cycles;

each available switch input priority list includes a sequential input pointer that, following the acceptance of a first nominating switch output by a first switch input, is incremented to a second switch input, next in sequence to the first switch input; and,

the arbiter nominates the available switch input closest in succession to the second switch input in subsequent arbitrations;

wherein the arbiter arbitrates between the nominating switch outputs in response to a switch input receiving multiple switch output nominations by accepting the least recently available nominating switch output as follows:

the arbiter accepts nominating switch outputs in response to a nominating switch output priority list, for each switch input receiving a plurality of nominating switch outputs;

each nominating switch output priority list includes a sequential output pointer that, following the acceptance of a nominating switch output, is incremented to a second switch output, next in sequence to the first switch output; and,

the arbiter accepts the nominating switch output closest in succession to second switch output in subsequent arbitrations.

44-53. canceled

54. (currently amended) The system of claim [[53]] 43 wherein the input pointer is incremented in response to the acceptance of the first nominating switch output by a first switch input, in the first arbitration cycle only.

55. canceled

56. (currently amended) The system of claim [[55]] 43 wherein the output pointer is incremented in response to the acceptance of the first nominating switch output by a first switch input, in the first arbitration cycle only.

57. canceled

58. (currently amended) The system of claim [[57]] 43 wherein the arbiter arbitrates between each switch output and a plurality of available switch inputs by arbitrating in a plurality of arbitration cycles, for each crossbar.

59. (original) The system of claim 58 wherein the arbiter nominates first available switch inputs, for each switch output, for a first crossbar;

wherein the available switch input priority list pointers are incremented to a second switch input, next in succession to the first switch input, following the acceptance of first available switch inputs;

wherein the crossbar counter is incremented to a second crossbar, next in succession to the first crossbar; and,

wherein the arbiter nominates switch inputs closest in succession to the second switch input for each switch output of the second crossbar.

60. (original) The system of claim 59 wherein the arbiter simultaneously arbitrates between a plurality of nominating switch outputs, sequentially for each crossbar.

61. (original) The system of claim 60 wherein the crossbar counter selects a first crossbar;

wherein the nominating switch output priority list pointers are directed to first switch outputs;

wherein the arbiter selects the first nominating switch outputs for the first crossbar;

wherein the arbiter accepts first nominating switch outputs;

wherein the nominating switch output priority list pointers are incremented to a second switch output, next in sequence to the first switch output, following the acceptance of the first nominating switch outputs;

wherein the crossbar counter selects a second crossbar, next in succession to the first crossbar; and,

wherein the arbiter accepts nominating switch outputs for the second crossbar closest in succession to the second switch output for each switch input.

62. (currently amended) An hierarchical arbitration system for transferring information across a switch, the system comprising:

a switch including a first plurality of crossbars having a plurality of parallel routed inputs and a plurality of parallel routed outputs connected to the switch inputs in response to the arbitration commands accepted on a control input;

an arbiter having an output connected to the switch control input to supply simultaneously arbitrated link commands for a plurality of links between the switch inputs and the switch outputs, for each crossbar;

a queue assembler having a plurality of inputs to accept variably sized information packets having a plurality of cells and addressing a plurality of outputs, and a control input to accept queue selection commands, the queue assembler grouping the information packets by switch output address, queuing the information packets into a plurality of queues for each address grouping, and supplying queues, selected in response to queue selection commands for each crossbar, at a plurality of outputs connected to corresponding switch inputs; [[and,]]

wherein the switch locks the links between switch inputs and switch outputs, in response to commands from the arbiter, to transfer information packets across the links;

wherein the arbiter includes a queue list with a queue pointer directed to a first queue, for each switch input, and selects the least recently available queue as follows:

in response to the queue pointer, the arbiter selects the first queue for each switch input accepting a nominating switch output for a first crossbar;

wherein the queue pointers, following the selection of each first queue, are incremented to a second queue, next in sequence to the first queue; and,

wherein the arbiter selects queues closest in succession to the second queue for each switch input accepting a nominating switch output for the second crossbar; and,

wherein the arbiter selects the least recently available queue for each crossbar.

63-64. canceled

65. (original) The system of claim 62 wherein the queue assembler accepts information packets having a ranked class of service (COS), and queues the information packets by COS; and,

wherein the arbiter selects a queue in response to COS of the information packet available for each crossbar.

66. (original) The system of claim 65 wherein the arbiter establishes a plurality of selection cycles per minor decision cycle, and simultaneously analyzes information packets at the head of each queue in each selection cycle.

67. (original) The system of claim 66 wherein the arbiter simultaneously analyzes the information packets at the head of each queue in response to the number of cells in each information packet.

68. (original) The system of claim 67 wherein the arbiter has an input to accept commands selecting an accumulation increment for each of the plurality of COS queues, where each accumulation increment defines a selected number of cells; and,

wherein the arbiter selects an information packet for transfer in response to the simultaneous analysis by comparing the number of cells in the information packet at the head of the queue to its corresponding accumulation increment.

69. (original) The system of claim 68 wherein the arbiter accepts commands selecting accumulation increments with larger numbers of cells for higher ranking COS queues.

70. (original) The system of claim 69 wherein the arbiter selects an information packet by comparing the number of cells in the information packet at the head of the queue to a corresponding total accumulation in a plurality of selection cycles.

71. (original) The system of claim 70 further comprising:

for each COS queue, a bank having a port connected to the arbiter for banking accumulation increments and supplying a banked accumulation;

wherein the arbiter, in each selection cycle, compares the number of cells in the information packets at the head of each COS queue to a total accumulation that includes the accumulation increment, plus the banked accumulation, as follows:

if an information packet has a number of cells less than, or equal to the total accumulation, making the information packet eligible for selection;

if information packets are eligible from a plurality of queues, picking the information packet in response to a priority system selected from the group including highest COS and the least recently used ordered service list;

if an information packet is picked, booking the total accumulation minus the number of cells in the selected packet in the bank corresponding to the selected packet COS queue; and,

if no information packets are picked, banking the total accumulation of each COS queue in its corresponding bank.

72. (new) In a switch system including a first plurality of crossbars with a plurality of parallel routed switch inputs and a plurality of parallel routed switch outputs, a hierarchical arbitration method comprising:

accepting variably sized information packets having a ranked class of service (COS) and including a plurality of cells at a plurality of switch inputs, the plurality of information packets addressing a plurality of switch outputs;

at each switch input, queuing the information packets by COS into a plurality of queues;

simultaneously arbitrating for a plurality links between switch inputs and switch outputs;

locking the links;

establishing a plurality of selection cycles;

simultaneously analyzing information packets at the head of each queue in each selection cycle;

selecting a queue for each locked link, for each crossbar, wherein each nominated switch input selecting a queue in response to COS of the information packets available for each crossbar; and,

transferring information packets across the links.

73. (new) An hierarchical arbitration system for transferring information across a switch, the system comprising:

a switch including a first plurality of crossbars having a plurality of parallel routed inputs and a plurality of parallel routed outputs connected to the switch inputs in response to the arbitration commands accepted on a control input;

an arbiter having an output connected to the switch control input to supply simultaneously arbitrated link commands for a plurality of links between the switch inputs and the switch outputs, for each crossbar;

a queue assembler having a plurality of inputs to accept variably sized information packets having a ranked class of service (COS) with a plurality of cells and addressing a plurality of outputs, and a control input to accept queue selection commands, the queue assembler grouping the information packets by switch output address, queuing the information packets into a plurality of queues for each address grouping

by COS, and supplying queues, selected in response to queue selection commands for each crossbar, at a plurality of outputs connected to corresponding switch inputs;

wherein the switch locks the links between switch inputs and switch outputs, in response to commands from the arbiter, to transfer information packets across the links; and,

wherein the arbiter selects a queue in response to COS of the information packet available for each crossbar, by establishing a plurality of selection cycles per minor decision cycle, and simultaneously analyzing information packets at the head of each queue in each selection cycle.